



Universidade Federal do Vale do São Francisco

# Estruturas de Dados

Professor: Marcelo Santos Linder

E-mail: [marcelo.linder@univasf.edu.br](mailto:marcelo.linder@univasf.edu.br)

## Ementa

- ➔ Alocação dinâmica de memória;
- ➔ Vetores, pilhas, filas, listas: representação e algoritmos para manipulação;
- ➔ Métodos de ordenação;
- ➔ Métodos de pesquisa;
- ➔ Noções de abstração de dados, de complexidade de algoritmos e de recursividade;
- ➔ Desenvolvimento, implementação e testes de programas, na linguagem de programação C, utilizando técnicas de estruturação de dados em aplicações particulares.

# Objetivos

## ➤ **Objetivo Geral**

- Possibilitar aos alunos a utilização otimizada das diversas estruturas de dados apresentadas, levando em consideração o problema a ser resolvido ou otimizado, e também o contexto no qual ocorre esse problema.

# Objetivos

## ➤ **Objetivos Específicos**

- Consolidar os conhecimentos sobre programação previamente adquiridos;
- Fornecer domínio da alocação dinâmica de memória;
- Apresentar as principais estruturas de dados e suas implementações, em termos de representação física e algoritmos de manipulação, guiando-se pelo conceito de tipos abstratos de dados;
- Introduzir aspectos básicos da complexidade de algoritmos;
- Prover uma introdução às técnicas de programação, com desenvolvimento, implementação e teste de programas, baseando-se no emprego de estruturas de dados em aplicações particulares;
- Apresentar os principais processos de pesquisa e classificação de dados.

## Metodologia

- ➔ A disciplina será trabalhada com aulas expositivo-dialogadas, onde serão fornecidos os componentes teóricos e será feita a prática de exercícios.

## Recursos

- ➔ Quadro branco, marcador, notebook e projetor multimídia.

## Forma de Avaliação

- A avaliação será realizada mediante 2 provas escritas e um trabalho prático.

## Bibliografia

### ➤ Bibliografia Básica:

- Estruturas de Dados Usando C – Aaron M. Tenenbaum, Yedidyah Langsam, Mosha J. Augenstein – Editora Pearson Makron Books – 2005
- Estruturas de Dados e Seus Algoritmos – Markenzon, Lilia – Editora LTC – 1994

### ➤ Bibliografia Complementar:

- C Completo e Total – Herbert Schildt – Editora Pearson Makron Books – 2006
- Algoritmos, Teoria e Prática – Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein – Editora Campus – 2002

## Informações Gerais

### ➤ Material de apoio

- Os slides utilizados em aula, listas de exercício e descrição de trabalhos encontram-se na página [www.univasf.edu.br/~marcelo.linder](http://www.univasf.edu.br/~marcelo.linder)

# Introdução



# Programa

Um programa pode ser visto como a especificação formal da solução de um problema. N.Wirth expressa em sua equação

*programa = algoritmo + estruturas de dados*

onde: o algoritmo contém a lógica do programa e os dados são organizados em estruturas de dados.

## Estruturas de Dados

A qualidade da solução de um problema depende, entre outros fatores, da forma como estão organizados os dados relevantes.

- Encontrar o número do telefone de um certo assinante no conjunto de todos os assinantes.
- Representação interna de uma string:

➤ 

4	G	A	T	O
---	---	---	---	---

;

➤ 

G	A	T	O	\0
---	---	---	---	----

.

Toda uma classe de modelos desenvolveu-se, ao longo do tempo, com o objetivo de viabilizar o processamento de dados.

## Estruturas de Dados

Estruturas de dados são formas genéricas de se estruturar informação de modo a serem registradas e processadas pelo computador.

Ex.:

- lista ordenada;
- vetores;
- árvores;
- grafos, etc.

Contudo estas só adquirem significado quando associadas a um conjunto de **operações**, que visam, de um modo geral, manipulá-las (algoritmos).

## Tipos Abstratos de Dados

Embora os termos “tipo de dado”, “estrutura de dados” e “tipo abstrato de dados” em essência se refiram aos mesmos objetos formais, eles são usados em contextos próprios.

O termo “tipo de dado” é usado no contexto de uma linguagem de programação (tipos primitivos) e está associado a um método de interpretar um padrão de bits.

## Tipos Abstratos de Dados

O termo “tipo abstrato de dado” (TAD) denota um modelo junto com um conjunto de operações definidas sobre o modelo. (tipo + operações validas).

Por fim, “estrutura de dados” é uma forma concreta de se implementar um TAD, ou seja, uma representação computacional do modelo matemático em questão.

## Tipos Abstratos de Dados

Para viabilizar a implementação de tipos abstratos de dados vamos nos valer das estruturas em C. Para exemplificar este conceito, definiremos um TAD RACIONAL.

Um número racional é o que pode ser expresso como quociente de dois inteiros. Definiremos a operação de criação e multiplicação de números racionais.

## Tipos Abstratos de Dados

```
#include <stdio.h>
typedef struct
{
    int num;
    int den;
}RACIONAL;
void criar_racional (int, int, RACIONAL *);
void multiplicar_racionais (RACIONAL *,
    RACIONAL *, RACIONAL *);
```

## Tipos Abstratos de Dados

```
void criar_racional (int n, int d, RACIONAL *r)
```

```
{
```

```
    r->num = n;
```

```
    (*r).den = d;
```

```
}
```

```
void multiplicar_racionais (RACIONAL *a,  
    RACIONAL *b, RACIONAL *c)
```

```
{
```

```
    c->num = a->num * b->num;
```

```
    c->den = a->den * b->den;
```

```
}
```



## Tipos Abstratos de Dados - Exercício

Implemente as operações de soma e verificação de equivalência entre elementos do TAD RACIONAL, definido anteriormente.